

structure

COMPOSITE[G] \*

feature -- Commands

add\_child (mc: G)

feature -- Queries

get\_children: LINKED\_LIST [G]

language

basic\_language

ASSIGNMENT +

feature -- Attributes

rhs: EXPRESSION

feature -- Commands

accept (v: VISITOR)

set\_expression (e: EXPRESSION)

feature -- Constructor

make (cn: STRING\_8; e: EXPRESSION)

feature -- queries

get\_expression: EXPRESSION

ATTRIBUTE\_DECLARATION +

feature -- Attributes

type: STRING\_8

feature -- Commands

accept (v: VISITOR)

feature -- Constructor

make (cn, t: STRING\_8)

feature -- Queries

get\_type: STRING\_8

BASIC\_LANG \*

composite\_language

ROUTINE\_DECLARATION \*

feature -- Attributes

para: ARRAY [TUPLE [pn: STRING\_8; pt: STRING\_8]]

type: STRING\_8

feature -- Commands

add\_assignment (a: ASSIGNMENT)

feature -- Constructor

make (rn, t: STRING\_8; p: ARRAY [TUPLE [pn: STRING\_8; pt: STRING\_8]])

feature -- Queries

get\_para: ARRAY [TUPLE [pn: STRING\_8; pt: STRING\_8]]

get\_para\_names: ARRAY [STRING\_8]

get\_para\_type (i: INTEGER\_32): STRING\_8

get\_type: STRING\_8

QUERY\_DECLARATION +

feature -- Commands

accept (v: VISITOR)

COMMAND\_DECLARATION +

feature -- Commands

accept (v: VISITOR)

CLASS\_DECLARATION +

feature -- Attributes

attr: LINKED\_LIST [ATTRIBUTE\_DECLARATION]

attr\_count: INTEGER\_32

command: LINKED\_LIST [COMMAND\_DECLARATION]

command\_count: INTEGER\_32

query: LINKED\_LIST [QUERY\_DECLARATION]

query\_count: INTEGER\_32

type\_correct: BOOLEAN

feature -- Commands

accept (v: VISITOR)

add\_attribute (a: ATTRIBUTE\_DECLARATION)

add\_command (rout: COMMAND\_DECLARATION)

add\_query (rout: QUERY\_DECLARATION)

set\_type\_correct (b: BOOLEAN)

feature -- Constructor

make (cn: STRING\_8)

feature -- Queries

get\_attr: LINKED\_LIST [ATTRIBUTE\_DECLARATION]

get\_command: LINKED\_LIST [COMMAND\_DECLARATION]

get\_query: LINKED\_LIST [QUERY\_DECLARATION]

COMPOSITE\_LANG [G] \*

LANGUAGE \*

feature -- Name

name: STRING\_8

feature -- Query

get\_name: STRING\_8

feature -- Visitor

accept (v: VISITOR)

expression

basic\_exp

INTEGER\_CONSTANT +

feature -- Attributes

value: INTEGER\_32

feature -- Constructor

make (num: INTEGER\_32)

require

num > 0

feature -- Queries

accept (v: VISITOR)

get\_value: INTEGER\_32

invariant

value > 0

NULL\_EXP \*

feature -- Attributes

value: STRING\_8

feature -- Queries

get\_value: STRING\_8

BOOLEAN\_CONSTANT +

feature -- Attributes

value: BOOLEAN

feature -- Constructor

make (b: BOOLEAN)

feature -- Queries

accept (v: VISITOR)

get\_value: BOOLEAN

CALL\_CHAIN +

feature -- Attribute

chains: ARRAY [STRING\_8]

feature -- Constructor

make (a: ARRAY [STRING\_8])

feature -- Queries

accept (v: VISITOR)

get\_chain: ARRAY [STRING\_8]

NIL\_EXP +

feature -- Commands

accept (v: VISITOR)

feature -- Constructor

make

CURRENT\_EXP +

feature -- Commands

accept (v: VISITOR)

feature -- Constructor

make

BASIC\_EXP \*

composite\_exp

COMPOSITE\_EXP [G] \*

feature -- Attributes

get\_value: STRING\_8

value: STRING\_8

BINARY\_ADD +

feature -- Attributes

value: STRING\_8 = "+"

feature -- Queries

accept (v: VISITOR)

BINARY\_DIV +

feature -- Attributes

value: STRING\_8 = "/"

feature -- Queries

accept (v: VISITOR)

BINARY\_MOD +

feature -- Attributes

value: STRING\_8 = "%"

feature -- Queries

accept (v: VISITOR)

BINARY\_MULT +

feature -- Attributes

value: STRING\_8 = "\*"

feature -- Queries

accept (v: VISITOR)

BINARY\_SUB +

feature -- Attributes

value: STRING\_8 = "-"

feature -- Queries

accept (v: VISITOR)

BINARY\_AND +

feature -- Attributes

value: STRING\_8 = "&&"

feature -- Queries

accept (v: VISITOR)

BINARY\_OR +

feature -- Attributes

value: STRING\_8 = "||"

feature -- Queries

accept (v: VISITOR)

BINARY\_EQ +

feature -- Attributes

value: STRING\_8 = "=="

feature -- Queries

accept (v: VISITOR)

BINARY\_GT +

feature -- Attributes

value: STRING\_8 = ">"

feature -- Queries

accept (v: VISITOR)

BINARY\_LT +

feature -- Attributes

value: STRING\_8 = "<"

feature -- Queries

accept (v: VISITOR)

BINARY\_OP \*

feature -- Commands

set\_left (e: EXPRESSION)

set\_right (e: EXPRESSION)

feature -- Queries

left: EXPRESSION

right: EXPRESSION

invariant

binary\_op: children.count = 2

UNARY\_OP \*

feature -- Commands

set\_exp (e: EXPRESSION)

feature -- Queries

get\_exp: EXPRESSION

invariant

children.count = 1

UNARY\_LOG\_NEG +

feature -- Attributes

value: STRING\_8 = "!"

feature -- Queries

accept (v: VISITOR)

UNARY\_NUM\_NEG +

feature -- Attributes

value: STRING\_8 = "-"

feature -- Queries

accept (v: VISITOR)

EXPRESSION \*

feature -- Queries

accept (v: VISITOR)

get\_parent: EXPRESSION

parent: EXPRESSION

set\_parent (e: EXPRESSION)